Final Project

Using Quantum to Calculate a Mixed-Strategy Nash Equilibrium in a Finite Two-player Game

Group: 10

Member: 蘇磐洋、張力元、林予康

工作分配(工作難以切割清楚，都有互相合作，大致如下)：

蘇磐洋：報告 section1-8、實驗(1)、實驗(3)、實驗(4)、實驗(5)、實驗(6)，主要設計程式架構、debug

張力元：報告 section 5、實驗(2) 、實驗(3)、實驗(4) 、實驗(6)，主要撰寫程式主體、找 Nash equilibrium、debug

林予康：報告 section 6、實驗(3)、實驗(4) 、實驗(5)、實驗(7)、實驗(8)，主要撰寫程式主體、實作 quantum circuit、debug

1. Abstract

Our final project is an exploration of quantum game theory, combining comprehensiveness and depth of this burgeoning field.

Computing a Nash equilibrium is an NP problem, and finding a Nash equilibrium is of great importance in a lot of applications. Using the quantum algorithm, in particular, the HHL algorithm [1] and Grover's algorithm [2], along with some classical techniques, may reduce the NP problem to polynomial time, which is the original topic and goal of our group.

However, after some try and error, we switched gears. We decided to explore the limit of the quantum game theory, not restricted ourselves to the two-player game and putting the problem of finding a Nash equilibrium aside.

This report is organized as follows. In section two, we will talk about the motivation for choosing this topic. In section three, we will give an example of a quantum game to illustrate how to combine these two fields. In section four, we present three approaches to deal with quantum game theory. In section five, we will give a brief introduction to the HHL algorithm, which is the basis of the classical approach. In section six, we elaborate on the concepts of quantum games, which is the basis of the quantum approach and the focus of the final project. In section seven, we will give what we have done in the final project. In section eight, we draw our conclusions.

2. Motivation

There are two reasons why we choose this topic. One is from an academic perspective, and the other is from a practical perspective.

(1) An Academic Perspective

Computing a Nash equilibrium is an NP problem, more precisely, a DDAP problem, but not an NP-complete one [3]. Thus, its study is valuable and indispensable from an academic point of view, and now, with the infinite potential of quantum algorithms, we think that the problem of computing a Nash equilibrium efficiently may have great breakthroughs.

In lecture #20 of [4], the problem of computing a mixed-strategy Nash equilibrium (MNE) in a finite two-player game is categorized as an FNP problem. According to [4], "FNP problems are just like NP problems except that, for 'yes' instances, we demand that a solution be produced." Interestingly, some FNP problems are guaranteed to have a solution, such as finding an MNE or prime factorization. These problems generally have two properties. Conceptually, they are easier problems in NP problems in that a solution always exists. However, they are not closely related to each other. Prime factorization is guaranteed to have a solution by the fundamental theorem of arithmetic, while the existence of an MNE is due to Brouwer's fixed point theorem.

(2) A practical perspective

In this interconnected world, communication technology and the communication industry are all the more important. When we are dealing with communication in our everyday lives, the problem of resource allocation comes into being, which is closely related to game theory. Therefore, we need to devise many systems based on game theory, such as bidding systems and auction systems, to maximize utilities.

The most important indicator of a good auction is DSIC, but most of the time, to carry out the optimal system requires very high complexities, perhaps exponential time [4]. Take Vickrey–Clarke–Groves (VCG) auction in lecture #7 of [4] for example. This auction entails a payment rule, which is an optimization problem, making the time complexity of the whole mechanism may be exponential. Thus, the reduction of the time complexity of computing a Nash equilibrium will be useful from a practical point of view.

After discussing the importance of computing an MNE, we now give some reasons why quantum is suitable for this problem and what quantum game theory can provide. Since few people have tried to use quantum theory to solve classical game theory, the literature survey for this problem is scarce. Thus, we will focus on why we should develop quantum game theory despite the existence of a classical one. The following is largely from [5].

(1) An Academic Perspective

According to [5], "Since it (game theory) is based on probability to a large extent, there is a fundamental interest in generalizing this theory to the domain of quantum probabilities." Also, we speculate that quantum theory may not only generalize classical game theory to quantum game theory but may also solve some of the problems of classical game theory. Thus, from an academic point of view, using quantum theory to solve some problems of classical game theory or, more aggressively, incorporating quantum theory and game theory is reasonable.

(2) The Selfish Genes

According to [5], "If the 'Selfish Genes' [6] are a reality, we may speculate that games of survival are being played already on the molecular level, where quantum mechanics dictates the rules." While this concept is novel, we have read [6] and couldn't associate game theory used in [6] with quantum mechanics. However, with the advancements of science and technology, we can't rule out the possibility that quantum games are being played on the molecular level.

(3) Quantum Communication and Quantum Information

According to [5], "whenever a player passes his decision to the other player or the game's arbiter, he communicates information, which—as we live in a quantum world—is legitimate to think of as quantum information."

In the lesson, we have learned quantum teleportation, superdense coding, and the BB84 Protocol. In general, quantum communication has the good properties of being able to transmit information that is a real number and the ability to resist eavesdropping.

The ability to transmit a real number can help generalize a classical game in that in a real classical game when we need to transmit a strategy, we can only use a finite number of bits to encode the strategy while in a quantum game, we can avoid this predicament.

Moreover, the ability to resist eavesdropping can be beneficial to games that require an unbiased third party, such as gambling. We will give a detailed explanation of quantum gambling later.

(4) Better Outcomes for Games

The prisoner's dilemma is a well-known game since the Nash equilibrium is not the optimal outcome. More precisely, two players playing their dominant strategies result in a situation that is the only outcome that is not Pareto efficient [7]. In [5], when a quantum strategy is allowed, the prisoner's dilemma is no longer a dilemma. The details will be presented later. This example gives us some hope that by allowing quantum strategies, there may be better outcomes (better payoffs for each player) for some games.

3. An Example of Quantum Game

Before delving into the nuts and bolts of quantum theory and game theory, let's give an example that uses the ideas of these two theories: quantum gambling [8].

Gambling needs a system to ensure the fairness of the mechanism. Traditionally, this is achieved by introducing a trusted third party. However, a trusted third party doesn't exist all the time. Thus, in [8], quantum gambling is presented to deal with this problem. The details of [8] are as follows.

(1) The Rules of the Game

Alice uses two quantum boxes, A and B, to store a particle. The quantum states of the particle stored in the two boxes are $|a\rangle$ and $|b\rangle$, respectively. Alice prepares the particle in a state and sends box B to Bob. Bob wins R coins (R > 0) in one of the following two cases: (1) Bob opens box B and finds the particle. (2) Bob doesn't find the particle in box B and asks Alice to send box A to him. Then, Bob detects the state Alice prepared is not the committed state $|\psi_c\rangle$ ($|\psi_c\rangle = \sqrt{(1-\gamma)}|a\rangle + \sqrt{\gamma}|b\rangle$). In any other case, Alice wins one coin.

Alice's strategy is to prepare a state $|\psi\rangle = \sqrt{(1-\alpha)}|a\rangle + \sqrt{\alpha}|b\rangle$, where α ($0 \le \alpha \le 1$) is a parameter controlled by Alice.

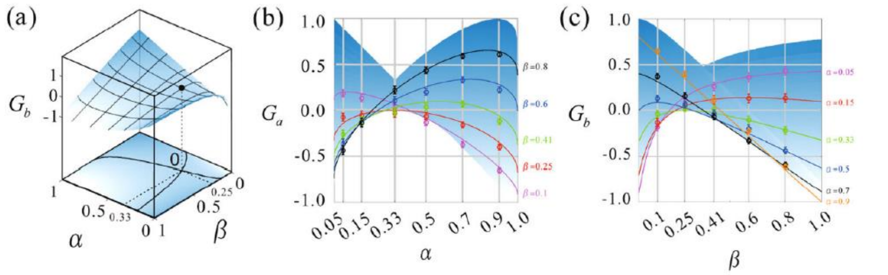
Bob's strategy is to split the received box B into two parts. One part is still called box B and the other one is a new box called B'. The state $|b\rangle$ becomes $\sqrt{(1-\beta)}|b\rangle + \sqrt{\beta}|b'\rangle$, where $|b'\rangle$ is the state in B' and β ($0 \le \beta \le 1$) is a parameter controlled by Bob.

(2) Proof for Both Honest Parties

If both Alice and Bob obey the rules without cheating, we can derive their best responses. There are many equations in [8], but the basic idea is simple. Follow the rule of the game and calculate the winning probabilities of Alice and Bob, respectively, and multiply the probabilities by the number of coins they can get, respectively. Finally, we can get their respective expected payoffs.

We give an intuition about the existence of a Nash equilibrium. If Alice chooses a small α, then Bob has a lower chance of finding the particle in the box. However, if α is too small, Alice will lose in the verification stage. The situation is similar for Bob since he needs to consider both stages, so his choice of β can't be too large or too small, either.

Following the above analysis, [8] gives the theoretical and experimental results in these figures for R = 1 and γ = 9/8, which is a fair zero-sum game. In (a), it can be seen that the marked point is a saddle point, meaning that this point is a Nash equilibrium.

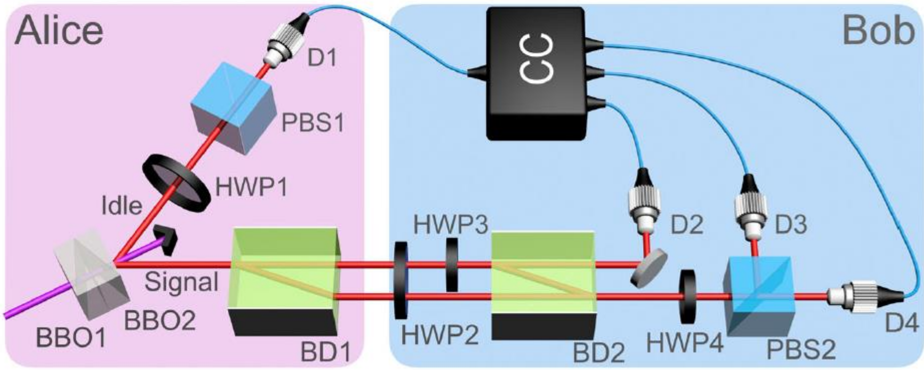


(3) Proof for Cheating

In [8], the proof is given in Discussion, Appendix A, and Appendix B. We won't give the details, but we will give some insight. Cheating involves manipulating the states or lying about the measurement results. If one party tries to manipulate the states, then the states may collapse, and the two-stage mechanism may be useful to catch the cheating party. If one party lies about the measurement results, the other one can measure the remaining state to testify whether the claimed results are correct.

(4) Experimental Demonstration

In [8], a proof-of-principle experimental demonstration is given using linear optics. HWP1 (half-wave-plate) is controlled by Alice and she can use HWP1 to determine α by rotating it. HWP3 is controlled by Bob and he can use HWP3 to determine β by rotating it. D2 (detector) is used to detect whether the particle is in box B. D3 and D4 are used to determine whether the transmitted state is equal to the committed state.



4. Our approach and related works

To combine quantum theory and game theory is not an easy task, as can be seen from the hefty papers and textbooks related to these two fields. However, on the other hand, we can also view this as an opportunity to delve into this area that has not yet been explored thoroughly.

There are three possible directions for us to calculate a mixed-strategy Nash equilibrium in a finite two-player game by using quantum theory. We will discuss the pros and cons of each of them and give the related works in the following.

(1) A Classical Approach

We can approach this problem in a classical way, meaning that the definitions of a game, Nash equilibrium, etc. won't be changed. This is a more realistic scenario in that classical games are likely to still prevail over quantum games in the foreseeable future. The difficulties lie in the interface between classical games and quantum circuits. Also, this approach is more restricted since we can't define games at our will. We will use the HHL algorithm [1] and Grover's algorithm [2] to accomplish this task.

(2) A Quantum Approach

We can approach this problem in a quantum way, meaning that the definitions of a game, Nash equilibrium, etc. will be re-defined thoroughly. This has the advantage of more freedom. However, to legitimately generalize classical game theory, we need to guarantee that the classical game is a special case in our quantum game, which is not trivial [5] [9].

(3) A Randomized Approach

We can approach this problem in a randomized algorithm, meaning that we will the algorithm is classical but we can use randomization, such as tossing a coin or randomly generating a number. This method is inspired by [10]. In [10], it uses a coin-tossing method, which can be easily achieved by a one-qubit circuit, to reduce the minimum spanning tree problem to expected linear time. We speculate that, as in the open problems in [10], randomized algorithms will prove useful in a lot of classical problems with the aid of simple quantum circuits.

5. A Classical Approach

The goal here is to find a Nash equilibrium fast to tackle the time complexity which needs $O(2^n * n^3)$ in the classical algorithm. More specifically, we know that for each strategy, we can choose to take or not to take it, which means there's a total of 2^n situations if we have n strategies. After finding the "correct sequence" of a Nash equilibrium, we also need n^3 to solve a system of equations of n variables. Hence, the total time is $O(2^n * n^3)$. (1 means taking the strategy, and 0 means not taking this strategy. If there are 5 strategies for both player 1 and player 2 and we know that the Nash equilibrium exists only when both players taking the first two strategies, then the sequence of the Nash equilibrium can be represented as 1100011000)

The implementation can be roughly divided into three parts. The first part is to implement the oracle, which produces the correct sequence of the Nash equilibrium directly. As for the second part, we intend to use Grover's search algorithm to acquire this sequence. First, flip the phase of this set, and then use reversion about mean. After going through several repetitions, we can get the sequence we want. Finally, use the classical algorithm to solve a system of equations of n variables $O(n^3)$. We expect that the method can largely shorten the time of finding a Nash equilibrium.

The details of the HHL algorithm are presented in "HHL_Algorithm.pdf," which is based on [1]. However, since we couldn't figure out how to use superposition to solve 2^n systems of equations simultaneously in the HHL algorithm, the quantum approach has become our focus.

6. A Quantum Approach

[5] provides a new way for us to deal with game problems. It views information as a physical quantity, rather than a mathematical entity. We will start our discussion from the simplest and symmetric game, the prisoner's dilemma.

For prisoner's dilemma in the classical domain, each player must independently decide whether she or he chooses to defect (strategy D) or cooperate (strategy C). The Nash Equilibrium is to both defects. That means both players couldn't have done better by unilaterally changing his or her strategy. However, is that the best we can do? Now, we turn to the quantum domain to explore the better outcome. In the end, we will show the following result. First, there exists a particular pair of quantum strategies which is a better Nash equilibrium. Second, there exists a particular quantum strategy where the player has a higher payoff if played against any classical strategy.

The first step is to transform the classical game into a quantum one.

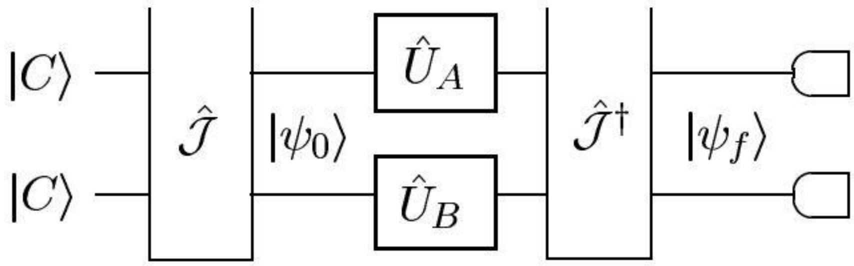
Model:

(1) a source of two bits, one bit for each player
(2) a set of physical instruments that enables the player to choose his quantum strategy
(3) a physical measurement device that determines the players' expected payoff from the state of the two bits

(all above are all known by players)

Possible outcomes of the strategies are assigned into the qubits in the quantum game, so the tensor product of the two qubits means the state of the game, where |DD>, |CC> |CD>, and |DC> are the bases.

The circuit of this game is described below.



The game starts from an initial state $\hat{J}$|CC>, where $\hat{J}$ is a unitary operator which is known to both players. Then, players will decide their strategies which are $\hat{U}_A$ and $\hat{U}_B$ chosen from a strategic space S. Finally, the measurement step consists of applying the inverse of $\hat{J}$ operator followed by measuring the outcomes on the classical game basis.

With the probability on each classical basis, we can calculate the players' expected payoffs according to their utility functions.

$$\$_A = rP_{CC} + pP_{DD} + tP_{DC} + sP_{CD}$$

Gate:

(1) U gate

U gate is the space from which players decide strategies. It proves to be sufficient to restrict the strategic space to the 2-parameter set of unitary 2*2 matrices, which is shown below.

$$\hat{U}(\theta, \phi) = \begin{pmatrix} e^{i\phi}\cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & e^{-i\phi}\cos\theta/2 \end{pmatrix}$$

$$0 \le \theta \le \pi \text{ and } 0 \le \phi \le \pi/2$$

θ governs the strategy played in the classical domain, while φ extends the classical domain into the quantum domain.

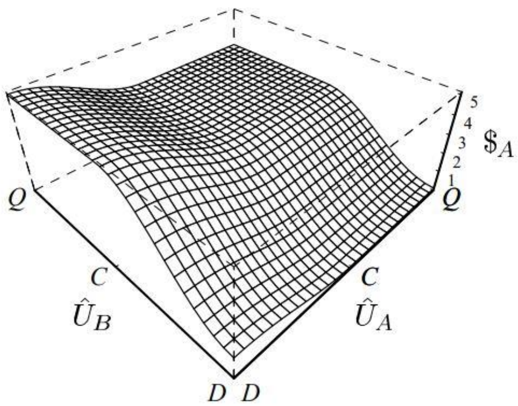
(2) J gate

A crucial role played by this operator is to introduce quantum entanglement, which we think of it as the dependent strategy between each player. Also, Ĵ needs to satisfy the following conditions. First, because the game is symmetric, the result of the interchange of two players must be the same for fairness. Hence, Ĵ needs to be symmetric. Second, the quantum game still needs to preserve its classical properties. Determining (1) and (2), we get Ĵ with a free parameter which decides how entangled the game is.

$$\hat{J} = \exp\{i\gamma\hat{D} \otimes \hat{D}/2\}$$

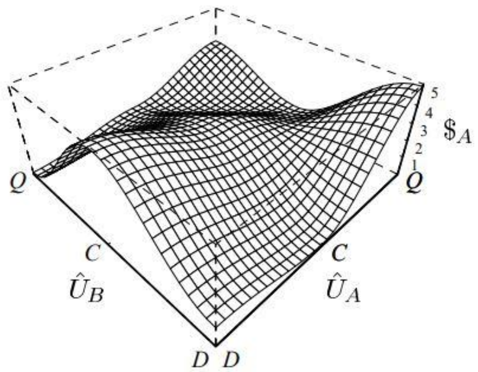
Discussion in cases:

(1) When γ=0, the game is separable. After plotting the 3D Graph of its expected payoff, we can find that playing D is the best response to each player. We can find "Defect" yields maximum payoff when making a cross-sector through any $\hat{U}_B$, the strategy of the other player. This separable game doesn't display any features which go beyond the classical game.



(2) When γ=π/2, the game is not separable, which implies that the strategy is out of the classical domain. In this particular case, if player A chooses

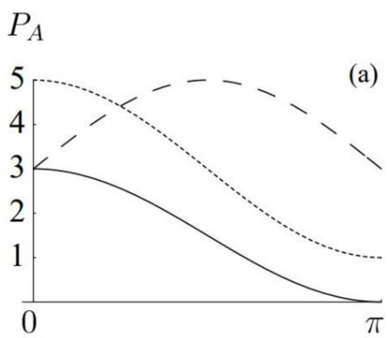
"defect," U(π, 0), the best response of player B would be Q, U(0, π/2). However, if player A chooses "cooperate," U(0,0), the best response of player B would be "defect," U(π,0). In this case, there's no Nash Equilibrium on both playing "defect".



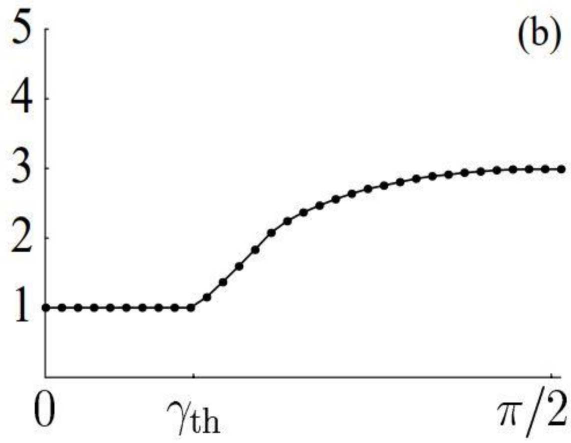
*Note that one must wonder what this kind of quantum strategy is under entanglement. We could explain the strategy as dependent on the other player's action so that player A will have a different strategy based on the action that player B takes.

Though both playing defects is not a Nash equilibrium, we could find that a new Nash equilibrium is generated while both playing Q and that no one can gain more utility by unilaterally deviating from (Q, Q). Also, the payoff for playing (Q, Q) now increases to (3, 3), which is the payoff of Pareto optimal outcome in the classical game. The major difference is that this is also a Nash equilibrium in the quantum game, indicating that quantum strategies allow the players to escape the dilemma.

Now, we turn to the unfair game which allows player A to use quantum strategies, while restricts player B to the classical domain. Note that the entanglement is maximum (γ=π/2) in this game. In this formation of the game, player A is advised to play U(π/2, π/2), where his payoff is at least 3. As for player B, his payoff can never be larger than 1/2. We could see the graph below. No matter which mixed classical strategy player A takes, the probability of getting payoff 1 is always positive. However, strategies in the quantum domain can successfully escape the circumstance and achieve a better outcome.



As we reduce the entanglement of the game progressively, we could find that the maxmin value decreases subsequently. When γ is $\sin^{-1}\sqrt{1/5}$, maxmin value becomes 1 so that player A can deviate and play "defect" instead, which produces the same result as that of a classical game.



In addition to the superior performance of the quantum strategies, the other benefit of a quantum game is to allow players to use "quantum" methods to transmit φ and θ of his strategy to avoid requiring lots of resources for dealing with real number transmission when implemented in a classical way.

Next, what if the number of players is more than two?

Only a small part needs to be revised in the circuit. First, we need N qubits for N players. Second, the J gate needs to be generalized. All we need to do is generalizing restrictions so that the quantum game also preserves its classical properties, which is to ensure commutators of $\hat{J}$ and different bases in the N-players game equal zero. It is found that after applying the restrictions, $\hat{J}$ can be expressed as the following in the maximal entanglement case [11].

$$\hat{J} = \frac{1}{\sqrt{2}}(\hat{I}^{\otimes N} + i\hat{F}^{\otimes N})$$

The most important property that a quantum game with more than two players is that there always exists a generalized pure strategy Nash equilibrium which has no analog in classical games, or even in two-player quantum games, which means we can reduce the time complexity when we try to obtain the Nash equilibrium.

In [11], a generalized pure strategy is presented. In traditional game theory, there is a fundamental distinction between so-called "pure" strategies, in which players choose their actions deterministically, and "mixed" strategies which can involve probabilistic choices. In the quantum domain, the distinction between a pure and mixed strategy is complicated. For example, U(π/2, π/2) should be viewed as a pure strategy while choosing U(π/2, π/2) with probability 1/2 and

U(π/4, π/2) with probability should be viewed as a mixed one. However, the pure strategy U(π/2, π/2) involves probability distribution among the actions, which is the property of a mixed strategy in the classical domain. Thus, in [11], they suggest that we should call these pure strategies in the quantum domain "coherent" strategies.

Let's look at an example of a quantum multiplayer game, the 4-player Minority Game. In the classical domain, the players have no better strategies than to choose randomly between the "0" and "1" actions. Then, the expected payoff for each player equals 1/8. However, we can maximize the expected payoff to 1/4 in a quantum game if each player takes the following strategy. It can be proved that this is the Nash equilibrium.

$$\left(\frac{1}{\sqrt{2}}(\hat{I}+i\hat{\sigma}_x), \frac{1}{\sqrt{2}}(\hat{I}+i\hat{\sigma}_x), \frac{1}{\sqrt{2}}(\hat{I}+i\hat{\sigma}_x), \frac{1}{\sqrt{2}}(\hat{I}-i\hat{\sigma}_y)\right)$$

In conclusion, we know that transforming games into the quantum domain can benefit a lot not only in payoffs but also in the reduction of resources. However, there are many aspects needed for improvement including time-consuming calculation and restricted forms of games, such as symmetry.

7. Our programs

We have implemented a lot of games to explore the limit of quantum game theory.

We have used the HHL algorithm and Grover's search to find the Nash equilibrium in the prisoner's dilemma. The oracle is built based on the knowledge of the game.

The other games are all in the quantum domain, listed below. Most of the games discussed below are presented in [12]. We have implemented the prisoner's dilemma for two players and three players. We have implemented a four-strategy two-player game based on the classical prisoner's dilemma, whose construction is not found in any paper we read. We have implemented the CHSH game, which is related to the Breidbart basis in lab 2. We have implemented the PQ penny flip game. We have implemented a Bayesian game.

We will talk about them one by one.

(1) Classical Prisoner's Dilemma without the HHL Algorithm

The program is "prisoner's_dilemma.ipynb" in the "classical approach" folder. We use the following encoding. 0, 1, 2, and 3 correspond to q0, q1, q2, and q3, respectively. (D, D) is the Nash equilibrium in this game, corresponding to (q3, q2, q1, q0) = 0101 of our encoding. 0101 means that the probability of Bob playing D is greater than 0 (q0 = 1), the probability of Bob playing C is zero (q1 = 0), the probability of Alice playing D is greater than zero (q2 = 1), and the probability of Bob playing C is zero (q3 = 0). We write a function "calculate_matrix(A, B, sequence, m, n)" to decide whether a given input is a Nash equilibrium. We use the built-in solver for systems of linear equations of NumPy. To deal with the truncation errors of the program, we use the concept of ε-Nash equilibrium and allow for a deviation of 10^{-10} of the optimal payoff.

		1	0
		Bob: *C*	Bob: *D*
3	Alice: *C*	(3, 3)	(0, 5)
2	Alice: *D*	(5, 0)	(1, 1)

The oracle is built by inspection, which outputs 1 only if the input sequence is "0101." After using Grover's search on the oracle, we put our result into "calculate_matrix" to check whether it is a Nash equilibrium. If so, we are done. If not, we run Grover's search again.

```
start checking
probability >= 0
False
start checking
probability >= 0
solution is correct
payoff is correct
([1.0, 0], [0, 1.0])
([1.0, 0], [0, 1.0])
total number until finding NE:  2
```

On average, we need to search four times before finding the Nash equilibrium, while the traditional approach needs to search nine times ($(2^2 - 1)^2$). This is caused by the quadratic speedup of Grover's search.

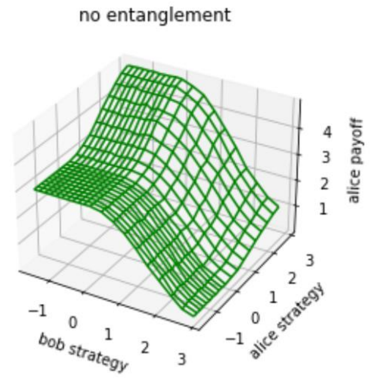
(2) Classical Prisoner's Dilemma with the HHL Algorithm

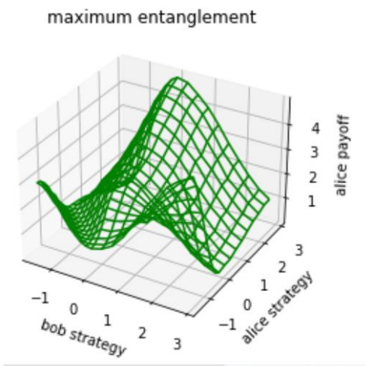
The program is "HHL.ipynb" in the "classical approach" folder. This program needs to be run on the IBM Q Cloud Server. The procedure is similar to (1) but uses the HHL algorithm to calculate the Nash equilibrium. The deviation is not small as listed below and the required resources are huge.

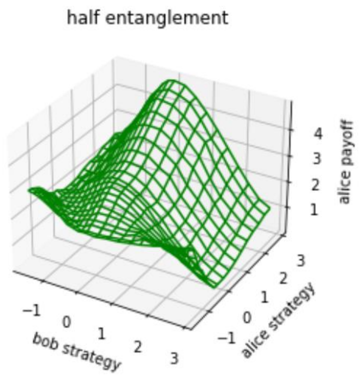
```
start checking
probability >= 0
payoff is correct

([(0.8682+0j), 0], [0, (0.86949+0j)])
```

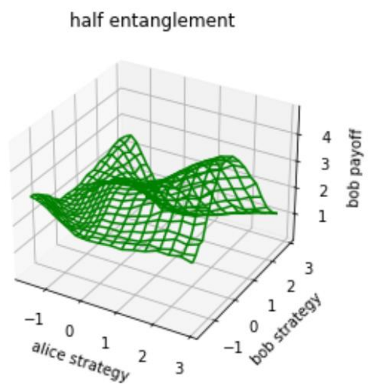
(3) Quantum Prisoner's Dilemma

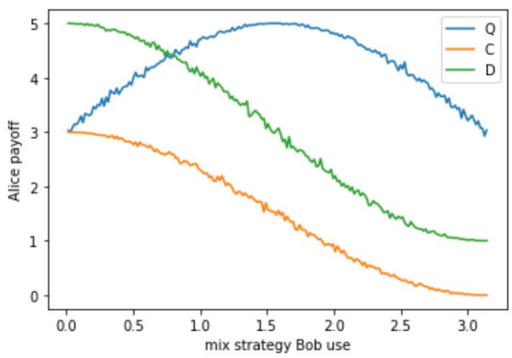
The programs are "2_player_game_graph.ipynb", "numerical_simulation.ipynb", and "2_player_symmetric_NE.ipynb" in the "2 player game" folder. The circuit is based on [5]. You can test the result with different φ and θ. We have drawn some figures to testify the results in [5]. Also, we have used a mathematical approach to simulate the results.











Finally, we have written a program to find the Nash equilibrium. Our approach is to try every combination of θ and φ. [12] has pointed out that [5] doesn't consider all the possible strategies. If we consider all the possible strategies, the parameters for each player should be θ, φ, and λ. Also, we need to consider mixed strategies. This is too complicated, so we restrict that each player's strategy is the following matrix as in [5]. We get the same result as [5].

$$\hat{U}(\theta, \phi) = \begin{pmatrix} e^{i\phi}\cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & e^{-i\phi}\cos\theta/2 \end{pmatrix}$$

```
******NE #1******
Player 1: theta = 0.0, phi = 1.5707963267948966
Player 2: theta = 0.0, phi = 1.5707963267948966
Payoff:  [3. 3.]
```

(4) Quantum Three-Player Prisoner's Dilemma

The programs are "3_player_game.ipynb" and "3_player_symmetric_NE.ipynb" in the "3 player game" folder. The circuit is based on [11]. You can test the result with different φ and θ. The following is the simulation result using the parameters in [11].

```
Nash Equilibrium:
player one payoff: 4.8984375
player two payoff: 9.0
player three payoff: 5.1015625
```

The Nash equilibrium is difficult to find out. If we divide the values of each parameter into 10 regions, corresponding to 11 points (0, π/10, 2π/10, ..., π), we will need to consider 11^8 points. That is, if we set n to be the number of regions for each parameter, the time complexity is $O(n^8)$. The program will run a long time, but we manage to find a symmetric Nash equilibrium. Note that we have restricted that each player can only change φ and θ. This setting is different from that in [11].

```
******NE #1******
Player 1: theta = 0.0, phi = 1.5707963267948966
Player 2: theta = 0.0, phi = 1.5707963267948966
Player 3: theta = 0.0, phi = 1.5707963267948966
Payoff:  [2. 2. 2.]
```

(5) Quantum Four-Strategy Two-Player Game

The program is "4_strategy_quantum_game.ipynb" in the "4 strategy game" folder. Different from [9], which uses a lot of parameters in a four-strategy game. We use two circuits of the prisoner's dilemma to simulate the game. Each player can manipulate two qubits. There are

2^4 possible outcomes, so the payoff matrix is 4*4. Although we may not entangle all the four qubits, we can greatly reduce the complexity of analyzing this circuit as in [9].

However, it is difficult to find the Nash equilibrium, either. Following the notation in (4), the time complexity is $O(n^8)$ as well. Therefore, in this program, we try to find the best response of a player when fixing the strategy of the other player. The time complexity is $O(n4)$. For example, given player 2's action and degree of entanglement as listed on the left-hand side, player 1's best response is listed on the right-hand side.

```
r1 = np.pi/2
theta2_1=0
phi2_1=-np.pi/2

r2 = np.pi/2
theta2_2=0
phi2_2=-np.pi/2
```

```
print(max_payoff)
```

```
2.8000000000000003
```

```
print("theta1_1: ", pi * index_i/10)
print("phi1_1: ", -pi/2 * index_j/10)
print("theta1_2: ", pi * index_k/10)
print("phi1_2: ", -pi/2 * index_l/10)
```

```
theta1_1:  0.0
phi1_1:  -1.413716694115407
theta1_2:  0.3141592653589793
phi1_2:  -1.413716694115407
```

(6) The CHSH Game

The program is "CHSH_game.ipynb" in the "other games" folder. The circuit is based on [12] and [13].

Unlike prisoner's dilemma, this is a cooperative game. The referee gives x to Alice and y to Bob. Both x and y are the Bernoulli(0.5) distribution. Alice returns a to the referee and Bob returns b to the referee. If a^b (a xor b) = x * y, they win. Otherwise, they lose.

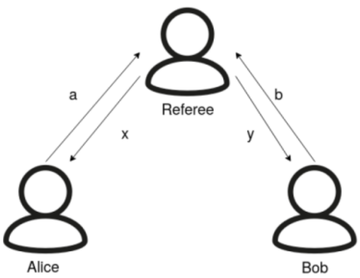


Figure 1: CHSH game

In the classical domain, the only action the players can do is to flip or not flip the bit, and the best winning probability is 0.75. In [13], a quantum strategy is proposed and the winning probability is about 0.8. In [12], a quantum strategy with a winning probability of 0.85 is proposed. However, when we use the method in [12], the winning probability is less than 0.4. We think that the strategy in [12] is not correct, so we made some modifications and the result is 0.85. This is similar to the idea of the Breidbart basis in lab 2. The results are listed below.

```
classical game payoff:  0.758
quantum game 1 payoff:  0.822
quantum game 2 payoff:  0.857
```

(7) The PQ Penny Flip Game

The program is "PQ_penny_flip.ipynb" in the "other games" folder. The circuit is based on [12] and [14].

Unlike prisoner's dilemma, this is a two-player zero-sum game. There are two players, P and Q. Q moves first, and then P, and then Q. In the classical game, each move can have two actions, flipping the penny or not flipping the penny. Initially, the penny is heads-up. If the result is still heads-up, Q wins. Otherwise, P wins. The winner has a payoff of 1, and the loser has a payoff of -1. The payoff matrix is shown below, which

	Q=NN	Q=NF	Q=FN	Q=FF
P=N	-1	1	1	-1
P=F	1	-1	-1	1

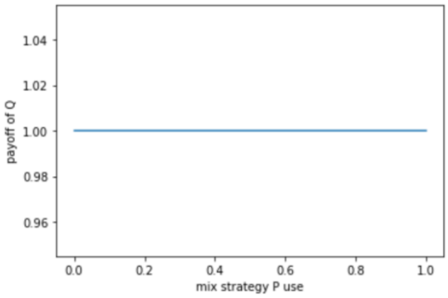
is from [12].

Now, if we can allow Q to have quantum strategies but not P, Q will win all the time. The quantum strategy allowed is on the left-hand side, and Q should play U(1/2, 1/2). P can only have two choices, F or N, listed on the right-hand side. After applying U(1/2, 1/2) to the initial state, the resulting density operator is the eigenstate of both F and N, so no matter P chooses F, N, or any combination of the two actions, Q will win. The result is shown below.

$$\mathrm{U(a,\ b)} = \begin{pmatrix} a & b^* \\ b & -a^* \end{pmatrix}$$

$$\mathrm{F} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \text{ or } \mathrm{N} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

```
We find in following graph that no matter what strategy P use, Q always win
Text(0.5, 0, 'mix strategy P use')
```



On the other hand, if we allow both of them to have quantum strategies, what will happen? We have demonstrated that a pure quantum strategy Nash equilibrium doesn’t exist, as stated in [14].

```
we can see that the best response of P can always achieve payoff=1
P best response: [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.
0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.
0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0,
1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0]

we can see that the best response of Q can always achieve payoff=1
Q best response: [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0]

no pure strategy Nash Equilibrium: []
```

[14] also proves that a mixed quantum strategy Nash equilibrium exists. However, the time complexity for finding a mixed quantum strategy Nash equilibrium is $O(2^{n*n*n})$. The first two “n”s correspond to Q’s actions, while the last “n” corresponds to P’s action. Therefore, we are not able to find the mixed quantum strategy Nash equilibrium due to the high complexity.

(8) The Bayesian Game

The program is “Bayesian_game.ipynb” in the “other games” folder. The circuit is based on [12] and [14].

Unlike all the games discussed above, a Bayesian game is a game with incomplete information, meaning that the payoff function of each player is not a common knowledge.

In [14], two payoff matrices are considered. Nature will determine which payoff matrix is used according a Bernoulli(p) distribution. You can change p at your will. For probability p, the payoff matrix is the upper one. For probability 1-p, the payoff matrix is a lower one. Thus, the players’ strategies are more complicated. They need to consider every possible case. In [14], this game experiments on a real device. However, we only simulate it but not test it on the real device to avoid noise.

$A\|B_1$	$\|0\rangle\,(C)$	$\|1\rangle\,(D)$
$\|0\rangle\,(C)$	$(11,9)$	$(1,10)$
$\|1\rangle\,(D)$	$(10,1)$	$(6,6)$

$A\|B_2$	$\|0\rangle\,(C)$	$\|1\rangle\,(D)$
$\|0\rangle\,(C)$	$(11,9)$	$(1,6)$
$\|1\rangle\,(D)$	$(10,1)$	$(6,0)$

The probability part is included in the final step where we calculate the expected payoff. To prevent the complexity of calculation of different strategies, the strategies provided in [14] is restricted to only I, X, Y, and Z operators. The uniqueness of the circuit in [14] is that it overlaps 8 strategies in one circuit. Hence, we only need two similar circuits to find all the outcomes. Nevertheless, after the implementation of the game, the Nash equilibrium we find is different from [14]. We aren't sure which one is correct.

```
Below is payoff matrix with four strategies
[11.  9.] [10.  1.] [11.  9.] [10.  1.]
[11.  9.] [10.  1.] [11.  9.] [10.  1.]
[1. 8.] [6. 3.] [1. 8.] [6. 3.]
[1. 8.] [6. 3.] [1. 8.] [6. 3.]

[11.  9.] is Nash Equilibrium without entanglement
```

8. Conclusions

We have implemented a lot of games in the final project. We try to explore how quantum mechanics can be used in game theory.

First, we take the classical approach. We manage to devise a new way to find a Nash equilibrium. Experiment (1) and (2) prove that our method is correct. However, the problem lies in the oracle. How to automatically generate the oracle given a payoff matrix is a great task worthy of future research.

Second, we have implemented the prisoner's dilemma for two and three players. When quantum strategies are allowed, the players can beat the dilemma. However, how to find a Nash equilibrium efficiently in this scenario is a big challenge.

Third, we have implemented a four-strategy two-player game. Traditionally, the literature tries to use a lot of parameters to control the degree of entanglement of the multi-strategy games. However, we take another approach. We separate the four-strategy two-player game into two two-strategy two-player games. Although this may not lose some degree of entanglement of the game, we think this is a perfect trade-off between realizability and performance. Can we use this method in games with an arbitrary number of strategies, not just a power of two? What are the properties of this kind of game? We think that these are good directions for future research.

Finally, we have implemented the CHSH game, the PQ penny flip game, and a Bayesian game. These are different types of games. We have demonstrated that when allowing quantum strategies, these games can have better performance than traditional ones. Also, we have pointed out the connection between the strategy in the CHSH game and the Breidbart basis in BB84, indicating that quantum game theory may have great use in the analysis of quantum communication. Can more types of games be implemented in the quantum domain? There are a lot more games that haven't been explored in a quantum perspective, such as the Stackelberg game, the Cournot model, the auction theory, the strong Nash equilibrium, etc. We think these are good topics for future research.

Both game theory and quantum mechanics are well-studied fields. These two fields will inevitably converge. When developing quantum game theory, we can not only generalize game theory but also gain more insights into the essence of quantum information.

This final project symbolizes an end as well as a beginning, signifying renewal as well as change. For quantum game theory, there is still a long way to go. But let's begin.

9. References

[1] https://qiskit.org/textbook/ch-applications/hhl_tutorial.html

[2] https://qiskit.org/textbook/ch-algorithms/grover.html and https://qiskit.org/textbook/ch-applications/satisfiability-grover.html

[3] https://people.csail.mit.edu/costis/simplified.pdf

[4] This is a thorough yet not-too-long lecture about auctions: http://timroughgarden.org/f13/f13.pdf.

[5] Jens Eisert, Martin Wilkens, and Maciej Lewenstein, "Quantum Games and Quantum Strategies"

[6] R. Axelrod, The Evolution of Cooperation (Basic Books, New York, 1984); R. Dawkins, The Selfish Gene (Oxford University Press, Oxford, 1976)

[7] https://www.cs.umd.edu/~nau/cmsc421/game-theory.pdf

[8] Pei Zhang, Xiao-Qi Zhou, Yun-Long Wang, Bi-Heng Liu, Pete Shadbolt, Yong-Sheng Zhang, Hong Gao, Fu-Li Li, Jeremy L. O'Brien, "Quantum

gambling based on Nash-equilibrium”
[9] Katarzyna Bolonek-Laso´n, “General quantum two-player games, their gate operators, and Nash equilibria”
[10] David R. Karger, Philip N. Klein, Robert E. Tarjan, “A Randomized Linear-Time Algorithm to Find Minimum Spanning Trees”
[11] Simon C. Benjamin, Patrick M. Hayden, “Multi-Player Quantum Games”
[12] Dhruv Devulapalli, Joey Brock, “Quantum Games and Entanglement as a Resource”
[13] Logan Meredith, “The CHSH game as a Bell test thought experiment”
[14] David Alan Meyer, “Quantum Strategies”
[15] Solmeyer, N. et al. (2018) Demonstration of a Bayesian quantum game on an ion-trap quantum computer. Quantum Science and Technology 3, 045002